

PowerShell 6 ▾

Version

- 6
- 5.1
- 5.0
- 4.0
- 3.0

Search

# Get-Alias

Module: [Microsoft.PowerShell.Utility](#)

Gets the aliases for the current session.

## In This Article

- [Syntax](#)
- [Description](#)
- [Examples](#)
- [Optional Parameters](#)
- [Inputs](#)
- [Outputs](#)
- [Notes](#)
- [Related Links](#)

PowerShell

Copy

**Get-Alias**  
[[<Name> <String[]>]  
[-Exclude <String[]>]  
[-Scope <String>]  
[<CommonParameters>]

PowerShell

Copy

**Get-Alias**  
[-Exclude <String[]>]  
[-Scope <String>]  
[-Definition <String[]>]  
[<CommonParameters>]

## Description

The **Get-Alias** cmdlet gets the aliases in the current session. This includes built-in aliases, aliases that you have set or imported, and aliases that you have added to your PowerShell profile.

By default, **Get-Alias** takes an alias and returns the command name. When you use the *Definition* parameter, **Get-Alias** takes a command name and returns its aliases.

Beginning in Windows PowerShell 3.0, **Get-Alias** displays non-hyphenated alias names in an <alias> -> <definition> format to make it even easier to find the information that you need.

## Examples

### Example 1: Get all aliases in the current session

PowerShell

Copy

PS C:\> Get-Alias

| CommandType | Name                  |
|-------------|-----------------------|
| -----       | ----                  |
| Alias       | % -> ForEach-Object   |
| Alias       | ? -> Where-Object     |
| Alias       | ac -> Add-Content     |
| Alias       | asnp -> Add-PSSnapin  |
| Alias       | cat -> Get-Content    |
| Alias       | cd -> Set-Location    |
| Alias       | chdir -> Set-Location |
| Alias       | clc -> Clear-Content  |
| Alias       | clear -> Clear-Host   |
| Alias       | clhy -> Clear-History |

This command gets all aliases in the current session.

The output shows the <alias> -> <definition> format that was introduced in Windows PowerShell 3.0. This format is used only for aliases that do not include hyphens, because aliases with hyphens are typically preferred names for cmdlets and functions, rather than nicknames.

### Example 2: Get aliases by name

| PowerShell                                       |  | Copy |
|--|--|------|
| <pre>Get-Alias -Name gp*, sp* -Exclude *ps</pre> |  |      |

This command gets all aliases that begin with gp or sp, except for aliases that end with ps.

### Example 3: Get aliases for a cmdlet

PowerShell

 Copy

```
PS C:\> Get-Alias -Definition Get-ChildItem
```

This command gets the aliases for the Get-ChildItem cmdlet.

By default, the **Get-Alias** cmdlet gets the item name when you know the alias. The *Definition* parameter gets the alias when you know the item name.

#### Example 4: Get aliases by property

PowerShell

 Copy

```
PS C:\> Get-Alias | Where-Object {$_.Options -Match "ReadOnly"}
```

This command gets all aliases in which the value of the Options property is ReadOnly. This command provides a quick way to find the aliases that are built into PowerShell, because they have the ReadOnly option.

Options is just one property of the AliasInfo objects that **Get-Alias** gets. To find all properties and methods of AliasInfo objects, type `Get-Alias | get-member`.

#### Example 5: Get aliases by name and filter by beginning letter

PowerShell

 Copy

```
PS C:\> Get-Alias -Definition "*-PSSession" -Exclude e* -Scope Global
```

This example gets aliases for commands that have names that end in -PSSession, except for those that begin with e.

The command uses the *Scope* parameter to apply the command in the global scope. This is useful in scripts when you want to get the aliases in the session.

## Optional Parameters

### -Definition

Specifies an array of aliases for the specified item. Enter the name of a cmdlet, function, script, file, or executable file.

This parameter is called *Definition*, because it searches for the item name in the Definition property of the alias object.

---

Type:  
String[]

---

Position:  
Named

---

Default value:  
None

---

Accept pipeline input:  
False

---

Accept wildcard characters:  
False

### **-Exclude**

Specifies an array of items that this cmdlet omits. The value of this parameter qualifies the *Name* and *Definition* parameters. Enter a name, a definition, or a pattern, such as s\*. Wildcards are permitted.

---

Type:  
String[]

---

Position:  
Named

---

Default value:  
None

---

Accept pipeline input:  
False

---

Accept wildcard characters:  
False

### **-Name**

Specifies the aliases that this cmdlet gets. Wildcards are permitted. By default, `Get-Alias` retrieves all aliases defined for the current session. The parameter name **Name** is optional. You can also pipe alias names to `Get-Alias`.

---

Type:  
String[]

---

Position:  
0

---

Default value:  
All aliases

---

Accept pipeline input:  
True (ByPropertyName, ByValue)

---

Accept wildcard characters:  
True

---

## **-Scope**

Gets only the aliases in the specified scope. The acceptable values for this parameter are:

- Global
- Local
- Script
- A number relative to the current scope (0 through the number of scopes, where 0 is the current scope and 1 is its parent)

Local is the default. For more information, see `about_Scopes`.

---

Type:  
String

---

Position:  
Named

---

Default value:  
None

---

Accept pipeline input:  
False

---

Accept wildcard characters:  
False

## Inputs

System.String

You can pipe alias names to **Get-Alias**.

## Outputs

System.Management.Automation.AliasInfo

**Get-Alias** returns an object that represents each alias. **Get-Alias** returns the same object for every alias, but PowerShell uses an arrow-based format to display the names of non-hyphenated aliases.

## Notes

- To create a new alias, use Set-Alias or New-Alias. To delete an alias, use Remove-Item.
- The arrow-based alias name format is not used for aliases that include a hyphen. These are likely to be preferred substitute names for cmdlets and functions, instead of typical abbreviations or nicknames.

## Related Links

- [About Aliases](#)
- [Export-Alias](#)
- [Import-Alias](#)
- [New-Alias](#)
- [Set-Alias](#)